

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2012, Article ID 951213, 10 pages
doi:10.1155/2012/951213

Research Article

Localization of Mobile Sensors and Actuators for Intervention in Low-Visibility Conditions: The ZigBee Fingerprinting Approach

Jose V. Marti,¹ Jorge Sales,¹ Raul Marin,¹ and Ernesto Jimenez-Ruiz²

¹ Computer Science and Engineering Department, Jaume I University, Avenida Vicente Sos Baynat s/n, 12071 Castelló de la Plana, Spain

² Department of Computer Science, University of Oxford, Wolfson Building, Parks Road, Oxford OX1 3QD, UK

Correspondence should be addressed to Jose V. Marti, vmarti@icc.uji.es

Received 23 December 2011; Revised 29 May 2012; Accepted 18 June 2012

Academic Editor: Hannes Frey

Copyright © 2012 Jose V. Marti et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Indoor localization in smoke conditions is one of the EU GUARDIANS project goals. When smoke density grows, optical sensors such as laser range finders and cameras cease to be efficient. Zigbee sensor networks provide an interesting approach due to the fact that radiofrequency signals are propagated easily in such conditions. Moreover, they permit having an alternative communication infrastructure to the emergency brigades, allowing also the implementation of localization algorithms for the mobile sensors, actuators, and firefighters. The overall localization method (i.e., ARIEL) aims to acquire the nodes position in real time during an intervention, using different sensor inputs such as laser, sonar, Zigbee, and Wifi signals. Moreover, a fine grained localization algorithm has been implemented to localize special points of interest such as emergency doors and fire extinguishers, using a Zigbee programmable high-intensity LED panel. This paper focuses on the Zigbee fingerprinting localization method used to obtain the position of the mobile sensors and actuators by training a database of radio signals for each scenario. Once this is done the proposed recognition method runs in a quite stable and accurate manner without needing any sophisticated hardware. Results compare the procedure with others such as KNN, and neural networks, demonstrating the feasibility of the method for a real emergency intervention.

1. State of the Art

Localization of mobile sensors and actuators is an active research field that becomes even more interesting and necessary in indoor applications such as fire emergency interventions, where the GPS is either not accessible or not practical to be used [1, 2].

First of all, some works use the laser range finder as a way to obtain the position of a mobile system in indoor environments [3, 4]. This solution is quite straight forward when the geometrical map of the building is well known, including the furniture. Other works focus on using visual landmarks to localize the mobile systems through vision cameras [5, 6]. These two alternatives are very accurate in situations of good visibility (e.g., nonsmoke conditions), although they are expensive to be implemented.

Moreover, in the sensor networks community, several interesting localization methods based on radio-frequency signals can be found, which can be transmitted in smoke

conditions. In fact, some techniques have recently been proposed for determining the position of mobile nodes by measuring this kind of signals, such as time of arrival (TOA), time difference of arrival (TDOA), angle of arrival (AOA), received signal strength (RSSI), and others [7–9]. In particular, the TDOA method can use a radio signal combined with a sonar. By measuring the difference in time of flight between the radio and the sonar signals, one can estimate the distance between the transmitter and the receiver in a very accurate manner [10, 11] although some extra work must be done to avoid the effect of reflections.

Radiofrequency allows the distance between transmitter and receiver to be calculated by measuring the RSS (Received Signal Strength) and applying to it the propagation/attenuation model represented by (1):

$$RSS = A * d^{-n}, \quad (1)$$

where A is the RSS at 1 meter from the transmitter, d is the distance between transmitter and receiver, and n is the

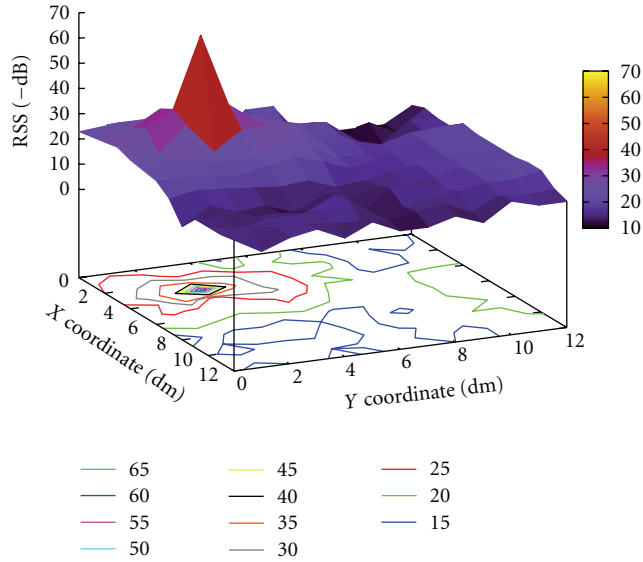


FIGURE 1: Signal strength distribution in an obstacle-free environment (outdoors).

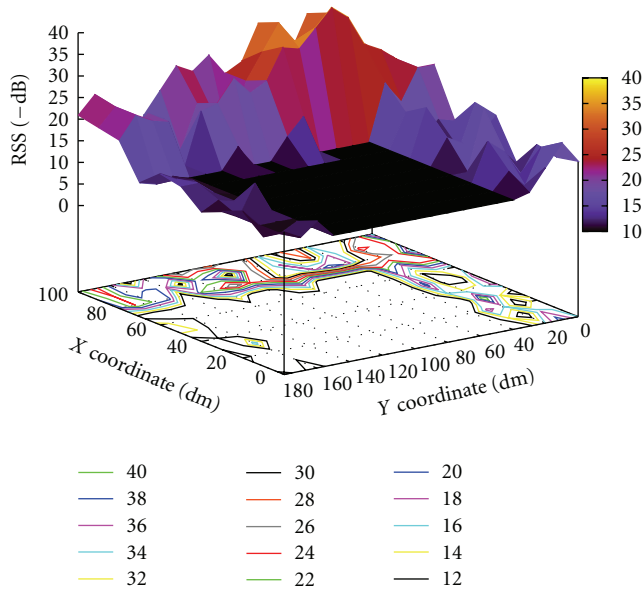


FIGURE 2: Signal strength distribution in an irregular environment (indoors).

propagation factor. In fairly open outdoor areas this is a suitable method to calculate distances, since there are no reflections nor interference and signal strength distribution is very clean, as shown in Figure 1. However, due to the unpredictable behavior of radio signals in indoor scenarios with irregular geometries and materials, other techniques must be studied, due to the fact that the RF behavior is affected significantly by these factors. For example, in Figure 2 the RF map is shown for a corridor that has stairs in the middle, including different kinds of metallic materials. The black area corresponds to the stairs hole, where the robot cannot be positioned and where it was not possible to take

any measures. Some methods such as RADAR [12] combine the empirical measurements and propagation model taking into account some geometrical characteristics of the environment, such as the presence of walls, improving the efficiency of the propagation/attenuation model. Other systems (see Youssef and Agrawala [13]) use probabilistic techniques, such as Bayesian estimation, to obtain the most probable transmitter position.

Fingerprinting methods consist of measuring the signal strength values to build a radio-frequency database model and then compare the navigation measures with those previously stored using pattern recognition techniques. These methods have the disadvantage of needing a previous training procedure for every location of a given scenario, and moreover, they adapt very well to the specific behavior of radio signals for a given space, which are affected by the particular characteristics of reflection, absorption, diffractions, and others, as explained in [14].

The *ZigBee* sensor network infrastructure is specially interesting for implementing fingerprinting localization methods, as it can be easily integrated in a building, offering many possibilities to control the radio signals characteristics such as power and frequency and enhancing the capacity of the trained radio map.

2. Introduction

In the frame of the EU GUARDIANS [15] project, a multi-sensor localization system has been developed in order to be able to obtain the localization of mobile robots and fire fighters inside a building during an intervention. The system, called ARIEL [16], uses different sensor inputs to calculate the positions (e.g., laser, sonar, *WiFi*, and *ZigBee* fingerprinting) and decides which one is the optimum at every moment depending on the environmental conditions (smoke density).

For example, when the smoke density is low, laser range finder sensors are still able to localize the nodes (Monte Carlo Localization method [17]), with a small positioning error of approximately 10 cm, once the building map is available and the structure of the building has not been affected. When the laser range finder detects a significant amount of smoke it considers it as an obstacle, so the *ZigBee* fingerprinting methods become a suitable alternative to have an approximate idea of the position, as we will see in the next sections.

Moreover, visual positioning based on visual servoing techniques [18] provide a fine-grain localization when the distance to the point of interest is reduced. For that, the ARIEL system provides a programmable *ZigBee* node that has a high luminosity LED panel attached on it, which can be perceived by the onboard camera in smoke conditions [16].

The present paper focuses on the radiofrequency localization method that has been implemented within the ARIEL system to obtain the nodes position in smoke-filled indoor areas. The paper compares several pattern recognition algorithms, in terms of efficiency and needed hardware complexity. Results show that the proposed fingerprinting method is

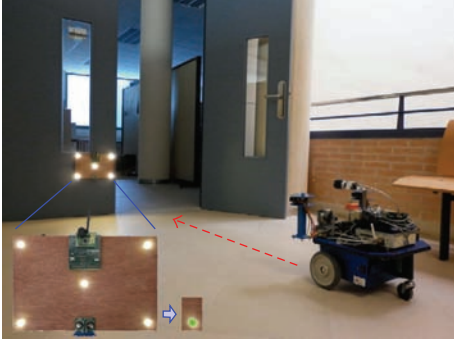


FIGURE 3: Mobile robot and the remotely controlled high-intensity LED panel.

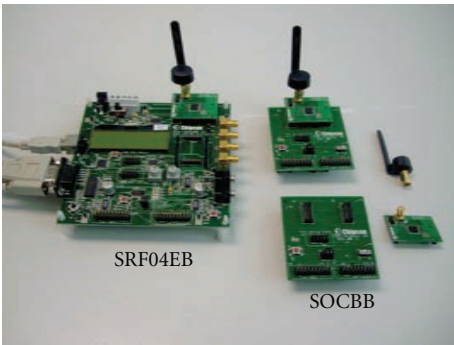


FIGURE 4: ZigBee communication modules.

TABLE 1: Channel-frequency matching.

Channel	Frequency
11	2405 MHz
13	2415 MHz
16	2430 MHz
19	2445 MHz
22	2460 MHz
26	2480 MHz

TABLE 2: Value-power matching.

Value	Gain
255	0.6 dB
95	-0.4 dB
19	-5.7 dB
3	-25.2 dB

suitable to be used in real interventions once the radio map for the given scenario is known through a training phase.

3. Hardware Description

The transceivers used are based on the CC2430 and CC2431 *Texas Instruments* microcontrollers and meet the *ZigBee* specification, with the capacity to obtain the RSS (Received Signal Strength) from every received packet. Moreover, 16

different channels can be configured with 256 different power levels. This fact has been used to increase the number of packets sent between the beacons and the mobile sensor at each robot position to improve the efficiency of the localization method.

On the other hand, the CC2431 microcontroller includes the *Location Engine* system that estimates the distance between each beacon and the transmitter by knowing the original signal intensity and the propagation coefficient of the medium. Then, by using three or more beacons, the system can triangulate the transmitter's position. This will allow us to compare the proposed fingerprinting localization method performance with the *Location Engine* mentioned above, as in [19], where it is easy to see that this method works well in open spaces but does not work properly indoors.

The experiments have been performed by using four transmitters in known positions (beacons) and one mobile transmitter, the position of which is going to be calculated. The whole sensor network information comes to one PC computer, carried by the robot, which calculates in real time the mobile transmitter position.

In summary, two different types of communication modules (nodes) shown in Figure 4 are involved in the measurements as follows.

- (i) SRF04EB (Serial Radiofrequency Evaluation Board): this board is going to be connected to a PC through a RS232 interface and will be used as a base station to send commands to the mobile transmitter and receive measurements from it.
- (ii) SOCBB (System On Chip Battery Board): this is the most simple board to hold a CC243X. They will be used for two possible functionalities.
 - (a) Mobile transmitter: this node will receive commands from the base station, perform the measurements, and send the results back to the base station (Figure 3).
 - (b) Simple beacon: there are four beacons located at fixed positions that simply return every packet received, including the RSS value.

4. System Training

In general, the proposed fingerprinting method works in two phases: training and localization estimation. In this section we will describe the experiments performed in order to obtain the measurements corresponding to the training phase and, in the next section, the ones used to calculate the transmitter position. Three different scenarios have been used: (1) garden (Figure 5), (2) classroom (Figure 6), and (3) corridor (Figure 7). Each of them has specific characteristics that will affect the signal propagation and RSS measurements.

The training procedure involves taking RSS measurements in different transmitter positions. In these measurements, beacons are placed at fixed positions, and the transmitter is located at every position of the scenario, using a certain density mesh (typically 50 cm by 50 cm). Then, data

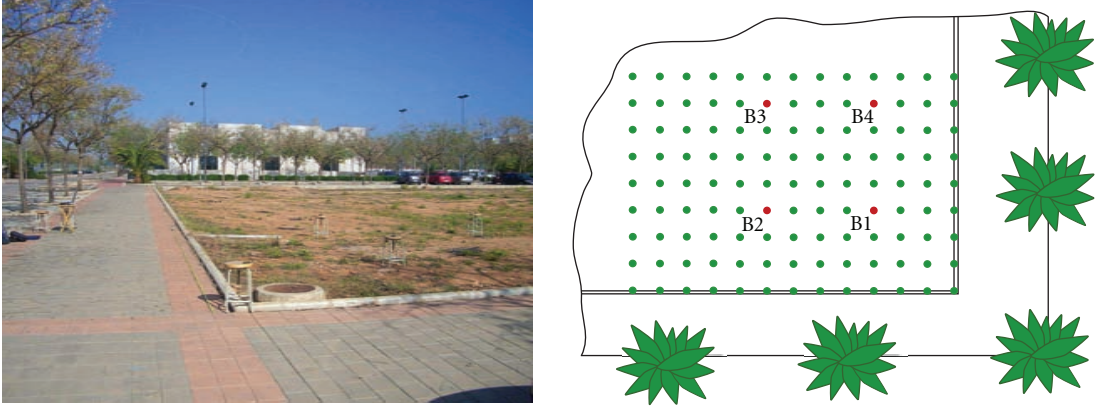


FIGURE 5: Scenario 1: Garden. B1, B2, B3, and B4 showing the beacons positions. Green dots are the different transmitter measurement positions.

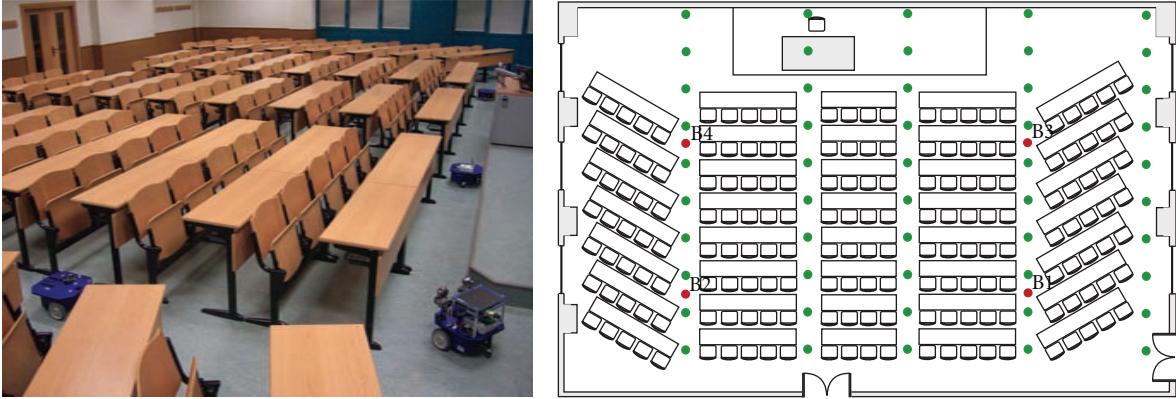


FIGURE 6: Scenario 2: classroom.

packet transmissions are made in different channels (frequencies) and using different power levels. In fact, for this experiment we used six channels and four power levels in order to cover the whole parameter range provided by the *Texas Instruments* transmitters used.

Specifically, for given used channels, the corresponding frequencies can be calculated with (2):

$$F = 2405 + 5 * (ch - 11) \text{ MHz}, \quad (2)$$

where ch is the channel number, which must take a value between 11 and 26. Then, channels 11, 13, 16, 19, 22, and 26, used in this experiment, correspond to the frequencies shown in Table 1. Also, the different power levels used can be seen in Table 2, where the first and last values are, respectively, the maximum and the minimum power the transmitter can generate.

The interference pattern distribution for these frequencies present a distance between nodes in the order of a few centimeters. Modifying the frequency of the transmitter will produce different interference patterns at the same transmitter location, as seen in Figure 8, and this will provide additional information to the location characterization measurements dataset.

For every combination of beacon, channel, and signal power, five packets are sent from the transmitter (mobile

sensor), which are sent back to the transmitter with the measured RSS value. This is done in order to have some statistical component in the data collected, avoiding spurious values.

To perform the training procedure the mobile sensor is placed at every position of the scenario (green dots in Figures 6 and 7), so that every RSS for every combination of beacon, channel, and signal power may be stored. The actual coordinates are also saved in the data base.

When a beacon receives a packet from the transmitter, it calculates its RSS and returns as confirmation a packet with a four-byte payload, as shown in Figure 9, where the beacon x and y coordinates in decimeters are sent in the first and second bytes. The third byte contains the beacon identification number and the fourth byte contains the obtained RSS value in—dB (i.e., a positive number between 0 and 90).

If a confirmation packet from the beacon is not received by the transmitter in a configurable amount of time, the transmitter sends a retry packet. This operation is repeated a configurable number of times. Finally, if no response is received, the transmitter sets the RSS to a minimum value of -99 dB for this particular combination of power, channel and beacon.

For every received packet, the transmitter measures the RSS and, with the beacon RSS, builds a pair of values that

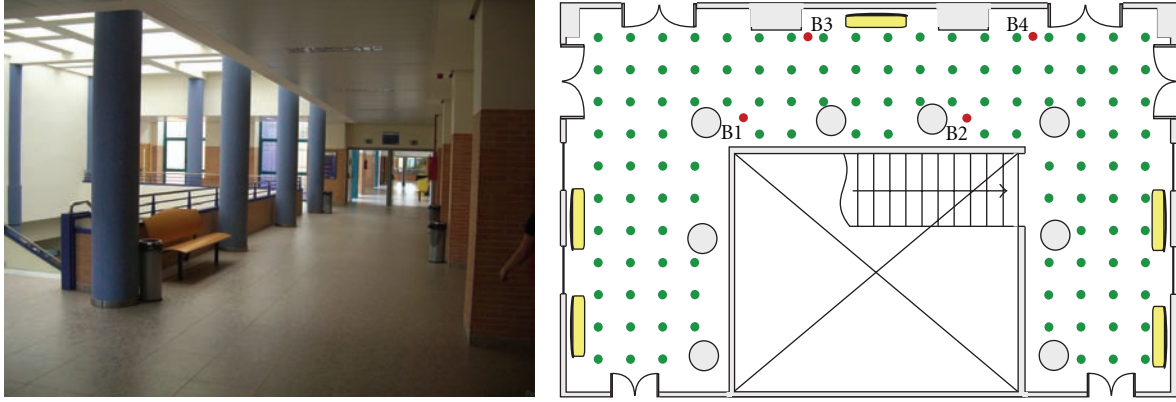


FIGURE 7: Scenario 3: corridor.

TABLE 3: Positioning error (decimeters) and calculation time (seconds). Results in *classroom* scenario.

Method	K-NN	MD	Neural	ARIEL	
			Network	Selected	Mean
Point 1	14.14	22.36	14.32	10.00	9.85
Point 2	133.42	58.31	14.42	0.00	17.20
Point 3	0.00	36.06	27.92	0.00	20.88
⋮	⋮	⋮	⋮	⋮	⋮
Point 53	20.00	14.14	59.06	10.00	8.00
Point 54	20.00	31.62	0.00	30.00	41.23
Point 55	20.00	160.31	76.59	20.00	12.65
Err Sum	1466.93	1841.40	1448.67	727.67	926.75
Err Mean	26.67	33.48	27.07	13.23	16.85
Std Dev	31.20	35.62	23.28	14.70	15.47
Calc Time	1.7 s	0.5 s	2.4 s	0.6 s	0.6 s

will be the measurement for this power, channel and beacon combination. For every transmitter position, six different channels and four power levels are used against four beacons. This represents a total of 96 couples of values (the one measured by the beacon and the one measured by the transmitter).

The transmitter collects the measures and forwards them to the base station, who will send it to the PC through the RS232 serial port. The PC adds to each packet the transmitter actual coordinates (previously introduced by hand as reference) and generates a new entry in the signal strength database. This information will contain the transmitter characterization for every position in the scenario.

Once the whole scenario has been measured, some calculations with the received data are made in order to condense the radio map. For every set of values obtained for each location, channel, power and beacon, a mean is calculated, reducing, with this procedure, the amount of information to a fifth. This is necessary to improve the system efficiency, considering that the aim is to obtain the robot localization in real time. The calculation time is then reduced from 8 s to 1,5 s. As system performance is critical in order to obtain a valuable localization procedure, working with the whole set of samples, as would happen when applying

any KNN-based algorithm, is not feasible. The ARIEL system provides this improvement, by enhancing the accuracy of the localization method and, at the same time, working with the simplified training set of radio samples.

5. Localization Estimation

Once the database is trained for a given scenario, the localization estimation procedure comes up, which consists of calculating the transmitter (mobile sensor) current position within the scenario. A mini PC in the robot stores the database and performs every calculation. Thus, the robot is completely autonomous in the localization aspect.

To accomplish this, the transmitter performs a set of measurements identical to those made in the training phase, with the corresponding channel, power, and beacon combinations. For that, the current RSS measurements set is compared with every sample stored in the database.

Several pattern recognition techniques have been compared in order to evaluate the performance of the ARIEL system.

5.1. Neural Network. Neural networks [20] have been successfully used for classification purposes (e.g., image

TABLE 4: Positioning error (decimeters) and calculation time (seconds). Results in *corridor* scenario.

Method	K-NN	MD	Neural Network	Selected	Mean
Point 1	31.62	189.74	19.66	30.00	30.08
Point 2	00.00	22.36	50.02	20.00	26.02
Point 3	00.00	22.36	14.13	31.62	26.93
⋮	⋮	⋮	⋮	⋮	⋮
Point 114	31.62	28.28	37.79	20.00	11.31
Point 115	10.00	14.14	22.49	22.36	11.40
Point 116	28.28	58.31	42.83	50.00	22.47
Err Sum	2895.58	4137.62	3657.60	1962.38	2301.44
Err Mean	24.96	35.67	31.53	16.92	19.84
Std Dev	27.76	39.46	29.84	10.38	23.45
Calc Time	3.4 s	1.3 s	3.8 s	1.5 s	1.5 s

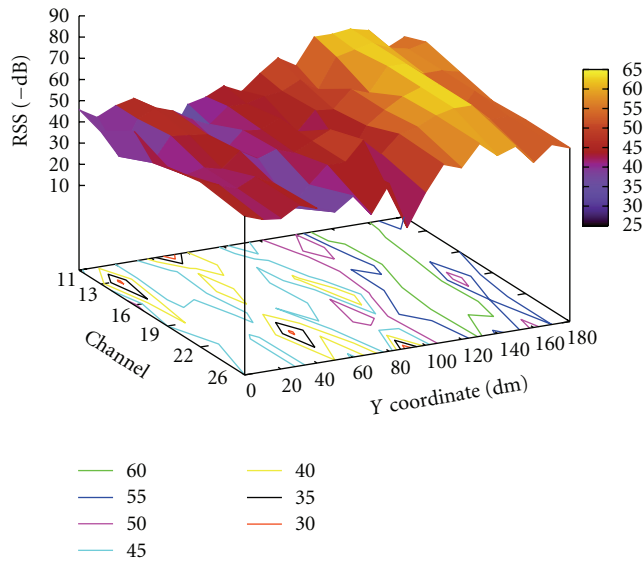


FIGURE 8: Signal strength distribution for different channels.

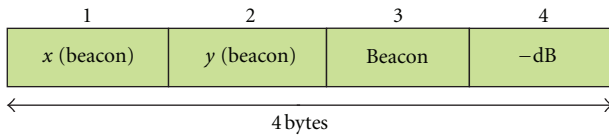


FIGURE 9: Beacon RSS measurement packet contents.

recognition [21, 22] or even in more sophisticated scenarios [23]).

In this paper neural networks have been used to estimate the position of the robot by taking as input the radio frequency inputs of a mobile node and the radio map that trains the network. For this, the *Resilient Backpropagation algorithm* [24] has been used, based on the results obtained in previous work [21].

In fact, the implemented neural network contains as many neurons in the last layer as available positions (i.e., 116

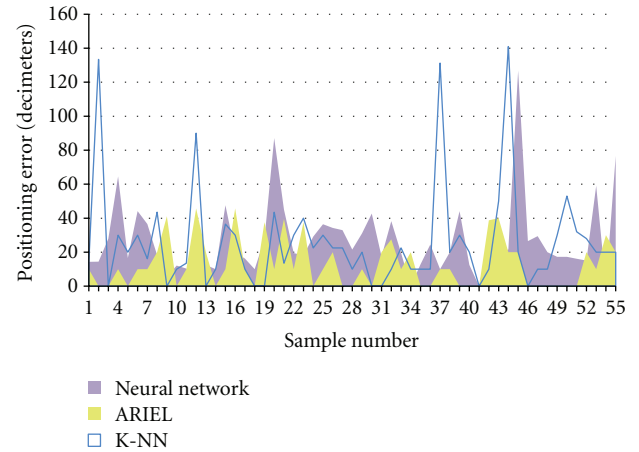


FIGURE 10: Results comparison between K-NN, Neural Network, and ARIEL selected methods in classroom scenario.

in the corridor scenario and 55 in the classroom scenario). Thus, each neuron will classify a given input parameter as a concrete position. For example, neuron 1 will be related with position $[0, 0]$. Experiments with several topologies and layers have been performed. Best results have been obtained using a 3 layers topology, with 100 neurons in the input layer and 200 in the hidden layer. Note that increasing the number of layers and number of neurons will not always lead to a performance improvement, since the error could be diminished when propagated through the network or by the creation of a local minimum; furthermore, the necessary time to converge to a solution also increments.

In the experiments, the whole set of 192 descriptors have been organized in groups of 4. Each of these subgroups represents the transmission/reception values for each beacon given a concrete configuration. Each descriptor group has been classified using a neural network with the above configuration. The position estimation has been calculated using the average of the output of each neural network for each subgroup of descriptors.

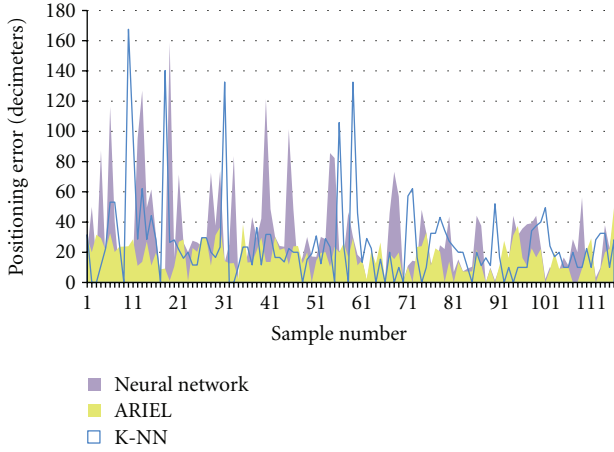


FIGURE 11: Results comparison between K-NN, Neural Network, and ARIEL Selected methods in corridor scenario.

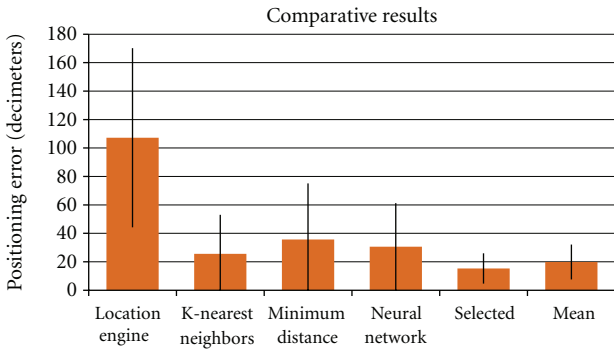


FIGURE 12: Results comparison between every method considered in corridor scenario.

5.2. ARIEL. The proposed method follows a similar criteria to the *k-nearest neighbors* pattern recognition method, where one calculates the *k*-nearest samples in the radio map that have a greater similarity to the sample obtained at the current mobile sensor position. Then, the recognition result is the more repeated position in this *k*-nearest vector.

Having this in mind, the following modifications have been implemented, in order to increase the whole system performance.

- (i) Once we have a RSS's sample (array) for the current position, we give more weight to the RSS values received by the beacons than the one calculated from the packets received by the transmitter, since the transmitter changes its signal power and beacon does not. Then beacons will receive different values for different power while transmitter will theoretically receive every confirmation packet with the same signal strength. There are two parameters (*wfb*: weight factor for beacon and *wft*: weight factor for transmitter) to adjust this.
- (ii) Two values do not need to be equal to be considered an RSS match. In fact, the parameter (*er*—equivalence radius) sets the maximum distance



FIGURE 13: Visual and ZigBee positioning experiments carried out at a paraffin smoke-filled small laboratory.

between two signal strength values to be considered identical.

- (iii) In addition to matches, for every couple of compared values (current measurement and database stored) the difference between them is calculated and stored. This value will provide extra information for recognition since the smaller this value, the better the match.
- (iv) As a result, after completing the comparison, eight candidates will be obtained (They are eight because it has been experimentally established that the correct transmitter position is between the eight best results the 95% of times) sorted by match and difference values. Depending on the matching level, the best candidate or the one with more candidate neighbors will be selected (as explained afterwards). To decide if two candidates are neighbors, the distance between them is calculated and then compared with the parameter *mnd*—maximum neighbor distance.

Then, for every transmitter position one will go over every RSS set stored in the database and calculate the two values (*matches* and *difference*). The matches (*M*) value will be obtained from (3):

$$M = \sum_{b=1}^4 \sum_{p=1}^4 \sum_{c=1}^6 \{ |\text{SSB}(b, p, c) - \text{CSB}(b, p, c)| < er \} * wfb + \{ |\text{SSB}(b, p, c) - \text{CST}(b, p, c)| < er \} * wft, \quad (3)$$

while the difference (*D*) value will be obtained by evaluating the equation (4):

$$D = \sum_{b=1}^4 \sum_{p=1}^4 \sum_{c=1}^6 |\text{SSB}(b, p, c) - \text{CSB}(b, p, c)| * wfb + |\text{SST}(b, p, c) - \text{CST}(b, p, c)| * wft,$$

where *b* is beacon id ($1 \dots 4$), *p* is power id ($1 \dots 4$), *c* is channel id ($1 \dots 6$), SSB is stored value for signal strength received by beacon, CSB is current value for signal strength received by beacon, SST is stored value for signal strength

```

# ADIF = Candidate A difference
# AMATCH = Candidate A matches
# BDIF = Candidate B difference
# BMATCH = Candidate B matches
MST = DST = 100 # Match and Difference Soft Thresholds
MHT = DHT = 110 # Match and Difference Hard Thresholds
diff = 100 BDIF / ADIF
match = 100 AMATCH / BMATCH
# If at least one of the hard thresholds is superated
# candidate A is better than candidate B
if ((diff ≥ DHT) or (match ≥ MHT))
{
  Candidate A better than candidate B
  end
}
# If both soft thresholds are superated
# candidate A is better than candidate B
if ((diff ≥ DST) and (match ≥ MST))
{
  Candidate A better than candidate B
  end
}
# If none of the previous conditions is satisfied
# candidate B is better than candidate A
Candidate B better than candidate A
end

```

PSEUDOCODE 1: ARIEL selected method pseudocode.

received by transmitter, CST is current value for signal strength received by transmitter, r is equivalence radius, w_b is weight factor in measures received by beacon, w_t is weight factor in measures received by transmitter, $A < B$: takes a “0” value if the expression is true and a “1” value if it is not

The next step consists of choosing the best candidate. From the sorted eight candidates list, if the first one (A) is much better than the second one (B), it will be considered the most probable transmitter localization. Two intermediate values are calculated to do this:

- (i) cm : Candidate A matches result respect candidate B matches result: $M(A)/M(B)$.
- (ii) cd : Candidate B difference result respect candidate A difference result: $D(B)/D(A)$.

In both cases, a higher value indicates a better result for the candidate A respect the candidate B. Four parameters are established as limits to decide:

- (i) CD_HARD and CM_HARD are limit values for cd and cm . Candidate A will be selected if ONE OF THEM is overcome by the calculated value.
- (ii) CD_SOFT y CM_SOFT are limit values for cd and cm . Candidate B will be selected if BOTH OF THEM are overcome by the calculated value.

In other words, if at least one of the two following conditions is accomplished, candidate A will be selected as the transmitter’s nearest location.

$$(cd > CD_{HARD}) \text{OR} (cm > CM_{HARD}), \quad (5)$$

$$(cd > CD_{SOFT}) \text{AND} (cm > CM_{SOFT}). \quad (6)$$

Otherwise, a two-dimension array called *dist* (distances between candidates) will be calculated with (7):

$$\text{dist}(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}, \quad (7)$$

where i and j are the array indexes, and x_k and y_k are, respectively, the x and y coordinates of the k th candidate. From this array, one list *numneighbors* is made to store the number of neighbors of every candidate. Two candidates are considered neighbors if they are closer than *mnd*, thus the array *dist* is searched for every candidate and one neighbor that added every time a value less or equal to *mnd* is found.

Once these calculations are made, the candidate with more neighbors will be *selected* as the best result. In Pseudocode 1 the equivalent pseudocode is shown.

As an additional method, a *mean* with the selected candidate and its neighbors coordinates is provided, with an extraweight (configurable in parameter *cp*—central point weight) for the selected candidate.

6. Experimentation Results

In previous works, the *Location Engine* engine, integrated into the *Texas Instruments* transmitters used has been compared with the exposed methods. In open spaces, as the *garden* scenario, the results are similar with respect to the localization error, and the calculation time, as expected, is hundreds of times faster in the analytical method, so there's no point to use the empirical methods into open spaces.

The exposed methods have been used to calculate the transmitter position in the two indoor proposed scenarios (classroom and corridor). Then, distances between the actual position and the one obtained by every method (i.e., the positioning errors) have been calculated, as well as the calculation time spent on every transmitter location estimated. From this information, sum, mean, and typical deviation for every scenario and method have been calculated. All these values are shown in the Table 4 for the *classroom* scenario and in Table 3 for the *corridor* scenario. The *garden* scenario results have not been included because outdoors environment and analytical methods work quite well and are easier to implement. On the other hand, in order to appreciate the ARIEL improvement, results of K-NN and Minimum Distance (i.e., MD) original methods have been included too.

Figures 10 and 11 show in a graphical way the results obtained with the three methods in the different scenarios. The ARIEL *Selected* method provides always better results than the neural networks used and, also, a more homogeneous error distribution.

Finally, Figure 12 shows the localization error results for every method considered.

7. Conclusions and Future Work

The paper has shown a proposed fingerprint algorithm for enhancing the efficiency of localization methods in indoor environments with irregular scenarios, including different materials. The ARIEL method increases the performance of several experimented pattern recognition methods such as K-NN, Minimum Distance, and Neural Networks and shows good results in every tested scenario. Combined with a designed high luminosity visual localization panel, the system may allow a robot to navigate in a smoky atmosphere and reach specific points of interest to help a firemen. Due to the complexity of filling with smoke the explored scenarios, some measurements have carried out in a small laboratory filled with paraffin smoke, as shown in Figure 13, showing no significant reduction in the precision. Further works will use real fire smoke.

It is necessary to consider that the neural network method requires a previous training phase for every given scenario and more hardware resources in the sensor nodes in order to perform the calculations. Future work will be focused on determining which measures give the most important information to the fingerprinting pattern recognition method, in order to reduce the amount of measurements involved, improving the calculation time and allowing the ARIEL method to be implemented with simpler hardware

devices. In the neural network aspect, more strategies need to be used in order to improve the recognition efficiency.

In addition, only the localization phase has been considered. In the navigation phase, once the position of the robot is reasonably known, only near positions will be searched in the database, both reducing the calculation time and the probability of significant errors in distance estimation.

Acknowledgments

This work has been funded in part by the EU-VI Framework Programme under Grant IST-045269-GUARDIANS of the EC Cognitive Systems initiative, the Bancaja-UJI research program under Grant RETA(P1-1B209-39), the Spanish National CICYT under Grant TIN2009-14475-C04, European Commission Seventh Framework Programme FP7/2007-2013 under Grant agreement 248497 (TRIDENT Project), by Spanish Ministry of Research and Innovation DPI2008-06548-C03 (RAUVI Project), and by Fundació Caixa Castelló-Bancaixa P1-1B2009-50.

References

- [1] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low-cost outdoor localization for very small devices," *IEEE Personal Communications*, vol. 7, no. 5, pp. 28–34, 2000.
- [2] S. Capkun, M. Hamdi, and J. Hubaux, "GPS-free positioning in mobile ad-hoc networks," in *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, p. 10, January 2001.
- [3] G. Cen, N. Matsuhira, J. Hirokawa, H. Ogawa, and I. Hagiwara, "Mobile robot global localization using particle filters," in *Proceedings of the International Conference on Control, Automation and Systems (ICCAS '08)*, pp. 710–713, October 2008.
- [4] M. Hentschel, O. Wulf, and B. Wagner, "A GPS and laser-based localization for urban and non-urban outdoor environments," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '08)*, pp. 149–154, September 2008.
- [5] D. Li, K. D. Wong, Y. H. Hu, and A. M. Sayeed, "Detection, classification, and tracking of targets," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 17–29, 2002.
- [6] S. Atiya and G. D. Hager, "Real-time vision-based robot localization," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 6, pp. 785–800, 1993.
- [7] D. Niculescu and B. Nath, "Ad Hoc positioning system (APS) using AOA," in *Proceedings of the 22nd Annual Joint Conference on the IEEE Computer and Communications Societies*, vol. 3, pp. 1734–1743, April 2003.
- [8] Z. Shan and T. S. P. Yum, "Precise localization with smart antennas in ad-hoc networks," in *Proceedings of the 50th Annual IEEE Global Telecommunications Conference (GLOBE-COM '07)*, pp. 1053–1057, November 2007.
- [9] G. Teng, K. Zheng, and G. Yu, "A mobile-beacon-assisted sensor network localization based on RSS and connectivity observations," *International Journal of Distributed Sensor Networks*, vol. 2011, Article ID 487209, 14 pages, 2011.
- [10] J. Sales, M. El-Habbal, R. Marin et al., "Localization of networked mobile sensors and actuators in low-visibility conditions," in *RISE (IARP/EURON Workshop on Robotics for*

- Risky Interventions and Environmental Surveillance*), Hallam University, Sheffield, UK, January 2010.
- [11] J. Sales, R. Mañá, E. Cervera, S. Rodríguez, and J. Pérez, "Multi-sensor person following in low-visibility scenarios," *Sensors*, vol. 10, no. 12, pp. 10953–10966, 2010, <http://www.mdpi.com/1424-8220/10/12/10953/>.
 - [12] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM '00)*, pp. 775–784, March 2000.
 - [13] M. Youssef and A. Agrawala, "The Horus WLAN location determination system," in *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services (MobiSys '05)*, pp. 205–218, June 2005.
 - [14] Q. Yao, F. Y. Wang, H. Gao, K. Wang, and H. Zhao, "Location estimation in ZigBee network based on fingerprinting," in *Proceedings of the IEEE International Conference on Vehicular Electronics and Safety (ICVES '07)*, pp. 1–6, December 2007.
 - [15] "GUARDIANS EU project (IST-045269) (Group of Unmanned Assistant Robots Deployed in Aggregative Navigation supported by Scent Detection)," <http://www.shu.ac.uk/research/meri/research/guardians-project>.
 - [16] J. Marti and R. Marin, "ARIEL: advanced radiofrequency indoor environment localization: Smoke conditions positioning," in *Proceedings of the 3rd International Workshop on Performance Control in Wireless Sensors Networks (PWSN '11)*, Barcelona, Spain, June 2011.
 - [17] Y. Wang, D. Wu, S. Seifzadeh, and J. Chen, "A moving grid cell based MCL algorithm for mobile robot localization," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO '09)*, pp. 2445–2450, December 2009.
 - [18] F. Chaumette and S. Hutchinson, "Visual servo control. I. Basic approaches," *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82–90, 2006.
 - [19] J. V. M. Aviles and R. M. Prades, "Pattern recognition comparative analysis applied to fingerprint indoor mobile sensors localization," in *Proceedings of the 10th IEEE International Conference on Computer and Information Technology (CIT '10)*, pp. 730–736, Bradford, UK, July 2010.
 - [20] G. P. Zhang, "Neural networks for classification: A survey," *IEEE Transactions on Systems, Man and Cybernetics Part C*, vol. 30, no. 4, pp. 451–462, 2000.
 - [21] E. Jiménez, R. Marín, and P. J. Sanz, "A soft computing classifier based on fourier descriptors within online robots context," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '04)*, pp. 4838–4843, October 2004.
 - [22] S. A. Nazeer, N. Omar, and M. Khalid, "Face recognition system using artificial neural networks approach," in *Proceedings of the International Conference on Signal Processing, Communications and Networking (ICSCN '07)*, pp. 420–425, February 2007.
 - [23] L. Shi, Z. Wang, L. Wang, and J. Zhang, "The aide diagnosis of cardiac heart diseases using a deoxyribonucleic acid based backpropagation neural network," *International Journal of Distributed Sensor Networks*, vol. 5, no. 1, pp. 38–38, 2009.
 - [24] M. Riedmiller and H. Braun, "Direct adaptive method for faster backpropagation learning: The RPROP algorithm," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 586–591, April 1993.